SKA 低频成像管线并行优化

韦建文[1]，张晨飞[1]，劳保强[2,3*]，林新华[1]，安涛[2]

1. 上海交通大学, 上海 200240
2. 中国科学院上海天文台, 上海 200030
3. 云南大学, 物理与天文学院, 昆明 650500
* 联系人, E-mail: lbq@shao.ac.cn

收稿日期: 2022-06-28; 接受日期: 2022-07-xx;
国家重点基础研究发展计划 (编号: 2018YFA0404600, 2018YFA0404603) 资助项目

摘要　平方公里阵列 (Square Kilometre Array, SKA) 射电望远镜的数据处理是通过管线方式进行的, 管线的执行效率是 SKA 区域中心考虑的重要因素. 连续谱成像观测是 SKA 的主要观测模式之一, 也是许多科学工作的基础. 本文以 SKA 低频先导设备 (Murchison Widefield Array, MWA) 的成像管线为例, 在中国 SKA 区域中心原型机 (China SKA Regional Centre prototype, CSRC-P) 上进行并行处理管线优化. 以往的优化方案都集中在少数性能热点, 缺乏对整体管线的系统优化, 导致整体加速比相对较低. 针对这一问题, 本文提出了一种全局优化方案, 针对管线使用多种编程语言、图像数据可独立处理的特点, 综合使用 C++ 多线程、Python 多进程、Shell 多任务并行等优化方法, 并验证了优化结果的准确性. 实验表明, 优化后的代码在 CSRC-P 的 x86 节点和 ARM 节点上分别获得了 2.7 和 2.4 倍加速. ARM 计算节点展现出对 SKA 应用良好的适应性. 本文的优化策略和方法也适用于其他 SKA 科学应用, 对 SKA 先导望远镜的科学运行和未来的运行有帮助.

关键词　平方公里阵列, 低频成像, 高性能计算, 并行优化

PACS: 47.27.-i, 47.27.Eq, 47.27.Nz, 47.40.Ki, 47.85.Gj

1　引言

为了支持不同的科学目标, 平方公里阵列 (Square Kilometre Array, SKA) 射电望远镜天文台将以不同的观测模式进行, 即成像模式和非成像模式. 其中, 连续谱成像和谱线成像是主要的两种成像模式. 在连续谱成像中, 天空区域可以在很宽的频率带宽上成像. 以该模式进行的射电巡天 (即: 连续谱巡天) 观测是 SKA 实现星系演化、宇宙大尺度结构演化、宇宙磁场等关键科学目标的重要前提 [1], 并将提供不同的科学数据产品, 包括连续谱和偏振图像 [2–4]、多波段星表 [5,6]、慢速瞬态搜索数据 [7] 和电离层测量 [8] 等.

在过去的几十年中, 研究人员在射电望远镜上进行了许多大型的连续谱观测, 例如: 20cm 暗弱图像射电巡天 (Faint Images of the Radio Sky at Twenty Centimeters survey, FIRST) [9]; 国家射电天文台甚大阵列巡天 (NRAO VLA Sky Survey, NVSS) [10] 和超大阵列巡天 (Karl G. Jansky Very Large Array Sky Survey, VLASS) [11]; 默奇森宽

视场阵列 (Murchison Wide-field Array, MWA) 的银河系及河外全天巡天 (GaLactic and Extragalactic All-sky MWA, GLEAM) [13] 和 MWA 拓展阵的银河系及河外全天巡天观测 (GaLactic and Extragalactic All-Sky MWA-eXtended, GLEAM-X; Natasha Hurley-Walker, 2022); 巨米波射电望远镜 (Giant Metrewave Radio Telescope, GMRT) 150 MHz TIFR GMRT 巡天 (TIFR GMRT Sky Survey, TGSS) [12]; 低频阵列 (Low Frequency Array, LOFAR) 的 2 m 巡天 (LOFAR Two-metre Sky Survey, LoTSS) [14]; MeerKAT 国际 GHz 银河系外巡天 (MeerKAT International GHz Tiered Extragalactic Exploration, MIGHTEE) 观测 [15]; 澳大利亚平方公里阵列探路者 (Australian Square Kilometre Array Pathfinder, ASKAP) 的宇宙演化图谱巡天 (Evolutionary Map of the Universe, EMU) [16] 等. 这些大型射电望远镜观测极大地促进了对射电天空的深入了解, 为其他射电天文观测项目奠定了基础, 促进了射电天文学的蓬勃发展.

作为 SKA 先导望远镜之一, MWA 开展了低频射电连续谱巡天观测, 即 GLEAM 项目. GLEAM 是迄今为止低频射电波段的最大规模观测之一, 覆盖了赤纬 +30 deg 以南的整个南部天区. GLEAM 的观测区域和频率范围与 SKA 低频阵列高度一致, 对未来 SKA 科学研究和数据处理有极高的参考意义. 除了建立观测星表 [17], GLEAM 还为广泛的科学研究提供基础观测数据, 包括射电星系和活动星系核 [19], 星系团 [22], 麦哲伦云 [23], 漫射银河发射 [18], 银河系的磁场 [20, 21], 银河系和银河系谱线 [24], 脉冲星 [25] 和宇宙射线. GLEAM 第一年 (2013 年 8 月 9 日至 2014 年 6 月 18 日) 和第二年 (2014 年 8 月 4 日至 2015 年 7 月 6 日) 的观测数据已经发布. 第一年的数据包括, 在 72-231 MHz 的频率范围内, 对赤纬 +30 deg 以南和银河纬度 10 deg 以上的天空区域进行的观测, 其中不包含一些复杂辐射的区域, 例如麦哲伦云. 第一年数据共覆盖约 24,800 deg^2 的天区, 按赤纬划分为 7 个条带, 每个条带观测 1 晚, 循环完成, 直至预定的天区全部扫描完毕. 经校准等处理后, 第一年所有快照观测的可见度 (visibility) 数据总量约为 300 TB.

为协助 SKA 数据处理及相关科研成果输出, 上海天文台推动中国 SKA 区域中心 (SRC) 的筹建, 并完成了中国 SRC 原型机 (China SKA Regional Centre prototype, CSRC-P) [26, 27] 及其软件系统 [28]. SKA 区域数据中心建成后, 将以 MWA 为基础, 同时开拓和 ASKAP、MeerKA0T 和 LOFAR 的合作, 开展 SKA 先导项目的科学预研究和技术发展, 其中包括低频射电脉冲星搜索研究 [29], 以及本文所研究的低频射电连续谱巡天观测项目. 目前, 利用 MWA GLEAM 观测项目的原始数据, CSRC-P 已被用于开展数据传输、存储、管线处理等关键技术的研究.

由于 SKA 先导采用的大视场、宽频带、高分辨率和多波束技术, 且观测持续很长时间, 海量观测数据的处理将会是一个巨大挑战 [30]. 一个典型的观测任务完成后将生成 PB 级的原始数据和数百 TB 级的图像数据. 因此, 我们使用管线系统来自动化处理 SKA 和 MWA 的数据. 通过将流程分解为多个步骤, 将不同步骤的操作重叠, 管线可以实现并行处理, 加快运行速度. 然而, 现有的成像管线效率低下, 无法充分利用超级计算机计算节点上的多核特性. 现有的优化方法只针对几个性能热点, 缺乏对整个系统的优化, 这导致相对较低的整体加速度. 我们需要在 MWA 现有管线的基础上探索更有效的优化方法, 以便高效地处理未来 SKA 的观测数据. 本文提出了一种针对 MWA 成像管线在 x86 和 ARM 处理器上的全局优化方法, 主要贡献包括:

- 将成像管线作为整体分析, 综合采用C++ 多线程并行、Python 多进程并行和 Shell 多任务并行, 进一步提高了原已并行运行的代码的运行效率, 在 x86 和 ARM 平台上获得了 2.7 和 2.4 倍的加速.
- 在 x86 和 ARM 平台验证了并行优化的可行性, 并验证了成像结果的准确性. 结果表明, 本文的并行优化方法适用于 x86 和 ARM 两种指令集不同的计算架构, 优化前后的结果差异在实验允许范围内, 满足 SKA 数据分析的精度要求.
- 从并行效率入手, 分析了管线并行度与并行效

率的关系, 指出了下一步的优化方向, 即单个 MWA 管线仍然无法发挥现代多核处理器的能力, 需要增加工作流协调机制, 同时处理多条管线.

本文其余部分安排如下: 第2章将在介绍 MWA 数据处理成像管线的基础上, 寻找程序热点, 有针对性地设计并行优化方法. 第3章讨论并行优化效果, 并对比优化前后的结果差异. 第4章中介绍了本领域的相关工作. 最后, 第5章概括了我们的工作, 并展望了将来的优化方向.

2 管线并行优化方法设计

2.1 低频成像管线

SKA 成像管线包含了对于 MWA GLEAM 数据处理的全过程 [17,31]. 图 1展示了 SKA 低频观测数据成像的流程图. 它包含五个阶段: ❶**预处理** (Pre-processing), ❷**建模** (Modeling), ❸**校准** (Calibration), ❹**成像** (Deep imaging), ❺**后处理** (Post-processing). 在❶预处理阶段, 管线调用 Cotter 软件 [33] 读入观测的快照原始数据, 搜索和去除数据中的射频干扰, 同时在时间和频率上对可见度数据做平均化处理, 使其时间分辨率为 4 s, 频率分辨率为 40 kHz. 最后产生通用天文软件应用程序 (Common Astronomy Software Applications, CASA) 的 Measurement Set(MS) 格式数据文件, 并使用 Dysco [32] 压缩数据. 在❷建模阶段, 管线比对星表数据库数据, 搜索在数据预处理阶段中已经去除射频干扰的图像的射电源, 制作出可被校准阶段处理的天空模型. 在❸校准阶段, 管线使用 MitchCal 算法 [34], 根据建模阶段得到的天空模型, 校准来自预处理阶段的数据. 校准后的数据将会被作为新的数据列 (通常为 ‘CORRECTED_DATA’ 列) 存储在 MS 文件中. 在❹成像阶段将调用大视场成像与反卷积软件 WCSClean 中的 *wsclean* [35] 命令对校准后的数据进行深度成像并获得快照图像, 主要参数见表1. 在❺后处理阶段, 管线运行 *fits_warp* 脚本 [36], 对斯托克斯 Stokes XX 和 YY 子带图像以及 Stokes I 子带图像进行流量密度标度和天体测量修正. 最后在快照图像上运行射电源搜索程序 Aegean [39], 将得到的源清单与 GLEAM 第 1 年公开的星表比对源的流量密度, 得到每个快照图像的流量密度标度校正系数. 在天体测量校正中, 校正步骤是通过对 Stokes I 30.72 MHz 子带图像中测得的源位置偏移进行径向基函数拟合, 在图像平面上比对 MRC [37] 和 NVSS [38] 星表以确定参考源的位置并进行最终的位置修正.

表 1 成像阶段使用的参数
Table 1 Deep imaging parameters

参数	取值
输出图像像素大小	4000×4000
最大洁化迭代次数	30, 0000
迭代阈值	σ
迭代掩膜值	3σ
洁化操作增益值	0.85
像素分辨率	23.43 arcsec
权重模式	briggs (“robust” = −1.0)
偏振参数	计算 4 个偏振方向: XX, YY, XY, YX
频率通道参数	分成 4 个子通道进行频率综合

2.2 管线热点分析

我们将 SKA 成像管线视为一个整体, 剖析了运行过程中的热点子函数, 根据其 “串行时间占比”, 由阿姆达尔定律估算了当前并行效率和并行优化潜力, 选择**运行时间长、与理论并行效率差距大**的函数进行优化. 应用程序在多机多核计算集群上, 可以利用任务的独立性, 使用不同的处理器核心并行处理任务, 从而缩短整体运行时间. 根据编写函数所用的计算机语言, 适用于 SKA 管线并行优化的并行编程方法包括: C/C++多线程并行、Python 多进程并行和 Shell 多任务并行. 我们将根据热点函数的特性选择优化方法.

成像管线在 x86 和 ARM 平台上的函数热点如图 2和图 3所示, *fits_warp*、*wsclean*、*shell_loops* 和 *calibrate* 是管线的函数热点, 分析如下.

fits_warp 是一个在❺后处理阶段用于校正图像的 Python 函数. 由于 *fits_warp* 的多个输入图像可以被独立处理, 且当前并行效率远低于理论值, 可采用 Python 多进程方法优化.

wsclean 是一个在❹成像和❺后处理阶段用于成像的C++程序. 当前的并行效率低于理论效率,

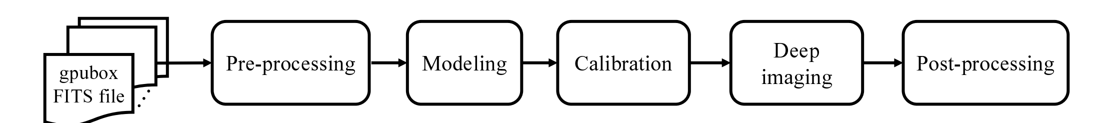


图 1 低频阵列的快照成像管线的流程图.
Figure 1 The snapshot imaging pipeline for low frequency array.

具有较好的 C++ 多线程优化潜力.

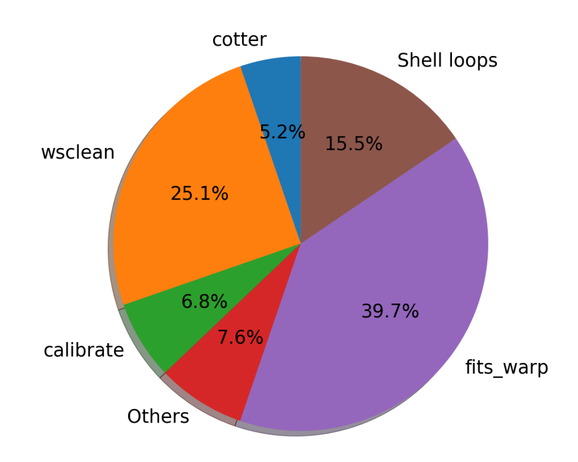


图 2 成像管线在 x86 节点上的热点
Figure 2 Hotspots of the imaging pipeline on a x86 compute node

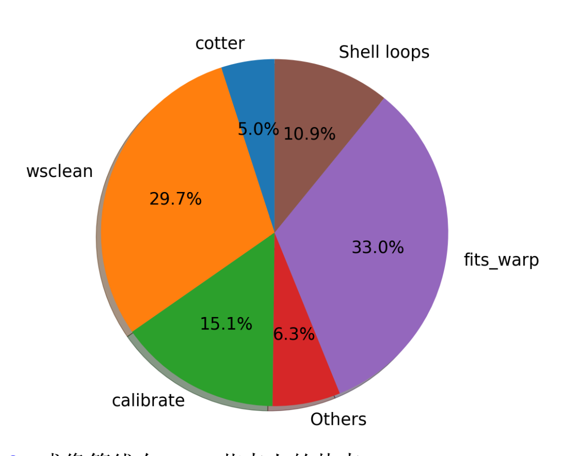


图 3 成像管线在 arm 节点上的热点
Figure 3 Hotspots of the imaging pipeline on a ARM node

calibrate 在❷建模、❸校准和❹成像阶段被调用, 用于宽场干涉快速成像. 其运行情况如表 2所示, 目前的并行效率为 72.5%, 较为接近理论并行效率 81.3%, 故不继续优化 *calibrate*.

shell_loops 是成像管线中的多层循环操作, 对不同天区的大量图像重复相似的处理流程. 原有 Shell 代码串行执行, 未发挥计算机的多核能力能力, 并行效率低, 可使用 Shell 多进程优化, 从而并行处理独立的图像来提高整体性能. 如表 2所示, 关于 *fits_warp*、*wsclean* 和 *shell_loops* 的详细优化将在下面的小节中介绍.

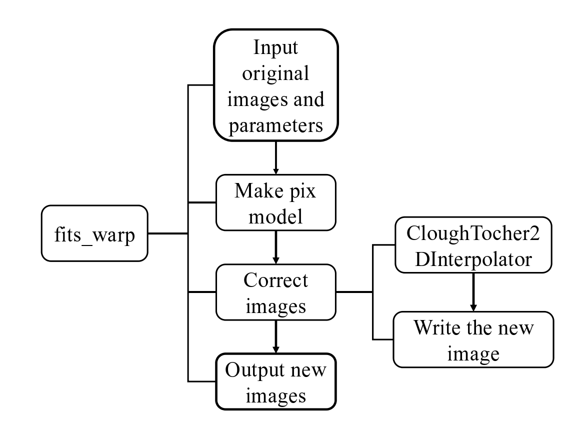


图 4 *fits_warp* 执行流程及热点
Figure 4 *fits_warp* 执行流程及热点

2.3 通过 python 多进程优化 *fits_warp*

fits_warp 的执行流程及热点如图4所示. 其中, *correct_images* 子函数是最耗时的部分, 该函数使用 *make_pix_model* 生成的校准模型来校准每幅图像. 而在 *correct_images* 中, *CloughTocher2DInterpolatorw* 占了主要的运行时间. 我们发现, 该函数中待处理的图像之间没有依赖关系, 可以在不同的进程中并行处理. 使用 Python multiprocessing 改写后的代码如下所示, 每个图像会在一个独立的进程中被校准, 由于每个进程会分配一个独立的 CPU 核心, 从而发挥多核节点的并行处理能力.

代码 1 使用 Python 多进程并行处理图像

```
def correct_images(fnames,  dxmodel,
    dymodel,  suffix):
```

表 2 针对热点函数的并行优化策略
Table 2 Optimization methods against hotspot functions

热点函数	调用阶段	当前耗时	当前并行效率	理论并行效率	优化策略
fits_warp	❸❹	2917s	8.3%	85.1%	Python 多进程
wsclean	❹❺	1609s	12.7%	65.5%	C++ 多线程和参数搜索
shell_loops	❹❺	1103s	-	-	Shell 多进程
caliberate	❷❸❹	485s	72.5%	81.3%	不做进一步优化

```
    ...
    for fname in fnames:
        multiprocessing.Process(target=
            parallel_func, args=[fname, x,
            y, xy, suffix]).start()

def parallel_func(fname, x, y, xy, suffix)
    :
    fout = fname.replace(".fits", "_"+
        suffix+".fits")
    im = fits.open(fname)
    data = np.squeeze(im[0].data)
    ...
    return
```

2.4 通过 C++ 多线程优化 *wsclean*

wsclean 在❷建模、❸校准和❹成像阶段被多次调用. 优化 *wsclean* 函数可以缩短这三个阶段的运行时间. 我们对 *wsclean* 进行了C++多线程优化和参数优化. 这些方法略微改变了原始 *wsclean* 代码中的算法, 因此我们在第3.4节中验证了优化代码的正确性. 我们的代码使用的是 *wgridder* 模式. 我们在源文件wsclean.cpp 和 wsmsgridder.cpp中, 对热点子函数 *runIndependentGroup* 进行 C++ 多线程优化. 在 wsclean.cpp中, 障碍函数griddingTaskManager->Finish() 被移到了上层循环, 使得多个predict()任务的执行不需要等待上一任务的完成. 在wsmsgridder.cpp中, 我们用每个线程的写入函数取代了聚合函数. 每个工作线程 (*predictCalcThread*) 都会并行地写到它的目标内存区域, 从而提高整体的吞吐量. 同时, 我们将超参数优化应用于两个 *wsclean* 参数--parallel-gridding 和--parallel-deconvolution. 前者控制 *imageMain* 和 *Predict* 子函数中的线程数量, 后者控制子图像的大小, 影响整体工作量和源匹配的准确性. 我们分别在 x86 和 ARM 节点上搜索这两个参数的最佳组合.

2.5 通过数据并行优化 *shell_loops*

MWA 成像管线由 Bash 脚本语言组织执行. 在❹成像和❺后处理图像中的四个循环显示了数据并行优化的潜力. 这四个循环分别记为L1-img, L2-img, L1-cor和L2-cor, 分散于 *image.sh* 和 *snapshot_correct.sh* 函数中. 我们使用 GNU Parallel [55] 工具改写原先串行执行的 Bash 脚本, 使其使用多进程并行处理多个独立的任务. 为了使任务负载与节点的 CPU 核心数相匹配, 我们尝试进行搜索, 以获得 GNU Parallel 工作进程数和调用程序线程数的最优组合.

具体地说, L1-img 在子通道、图像及偏振方向上进行迭代, 调用了子函数 *make_beam*, *pbcorrect* and *render* 来生成偏振图像. 由于任务之间没有数据依赖性, 我们先生成一个任务列表, 然后并行执行. 优化后的代码如代码 2所示, 使用了与 CPU 核数相等的工作进程数 (*NCPU*).

代码 2 L1-img 中的 Shell 多进程优化

```
# Generate a job list for L1-img
for subchan in MFS 0000 0001 0002 0003; do
  for image in $images; do
    for pol in XXr XXi XYr XYi YXr YXi YYr
        YYi; do
      echo "$subchan-$img-$pol" >> jobs.
          txt
# Run jobs in parallel
cat jobs.txt | parallel -j $NCPU make_beam
cat jobs.txt | parallel -j $NCPU pbcorrect
cat jobs.txt | parallel -j $NCPU render
```

L2-img在偏振方向、图像和子通道上迭代, 调用子函数 *rms_measure*、*pyhead* 和

copy_metafitsheader 来记录和重命名图像. 与L1-img类似, 我们展开这个 3 层循环, 使其并行运行, 并启动 *NCPU* 工作进程来并行处理独立数据, 在此省略代码示例.

L1-cor迭代 *make_beam* 和 *mv*, 对图像进行流密度校正. 我们展开这个 2 层循环, 然后启动 8 个 (4×2) 工作进程来并行处理.

代码 3 L1-cor 中的 Shell 多进程优化

```
# Generate a job list for L1-cor
for ((i=1; i<=(4); i++ )); do
  for pol in XX YY; do
    echo "$i-$pol" >> jobs.txt
# Run jobs in parallel
cat jobs.txt | parallel -j 8 make_beam
cat jobs.txt | parallel -j 8 mv
```

L2-cor 迭代 *agean*、*BANE*、*pyhead* 和 *correct_gleam_flux* 函数, 通过将图像中检测到的源与 GLEAM IDR6 目录交叉匹配来校正快照图像的流量比例. 子函数 *BANE* 和 *aegean* 已经是多线程的了, 为了配合 CPU 核心的总数, 其使用了较少的进程数. 伪代码如下所示.

代码 4 Multiprocessing in 'L2-cor'

```
# Generate a job list for L2-cor
for ((i=1; i<=(4); i++ )); do
  for file in ${Xcorr[$i]} ${Ycorr[$i]};
    do
    echo "$i-$fil" >> jobs.txt
# Run jobs in parallel
cat jobs.txt | parallel -j $(NCPU/NBANE)
    --cores=$NBANE
cat jobs.txt | parallel -j $(NCPU/NAEGENE)
     aegene --cores=$NAEGENE
cat jobs.txt | parallel -j $NCPU pyhead
cat jobs.txt | parallel -j $NCPU
    correct_gleam_flux
```

我们在管线上采用的优化策略汇总在表 2中. 对于 *fits_warp* 我们使用了 Python 多进程方法; 对于 *wsclean* 我们应用了C++ 多线程优化和参数搜索优化方法; 对于image.sh和 snapshot_cotter.sh 内的 *shell_loops*, 我们使用了 Shell 多进程优化方法; 我们没有额外优化 *caliberate* 函数.

3 结果与讨论

我们运行相同的算例, 从 "加速比" 和 "并行效率" 来评估管线优化结果, 并对比了 x86 和 ARM 平台的效果优化. 其中, "加速比" 定义为优化前与优化后运行时间的比值, 用来表示相对原程序的加速效果, 计算公式如下:

$$\text{Speedup} = \frac{t_{before_opt}}{t_{after_opt}}$$

"并行效率" 是程序使用单核心运行的时间与优化后运行时间的比值, 再除以分配的核心数, 用来表示程序使用多核心资源的效率, 上限为 100%, 计算公式如下:

$$\text{ParallelEfficiency} = \frac{t_{serial}}{t_{opt}\times N_{CPU}}$$

实验是在中国 SKA 区域中心原型机 (CSRC-P) [26] 上完成的, 其单节点的软硬件配置如表 3所示, 计算节点分别使用了 x86 和 ARM 指令集架构的处理器.

输入数据来自 GLEAM 快照数据, 观测 ID 为 1060179576, 观测时间为 UTC 时间 2013-08-10 14:19:18 至 2013-08-10 14:21:10, 观测相位中心在 (RA=290.51°, DEC=−26.73°).

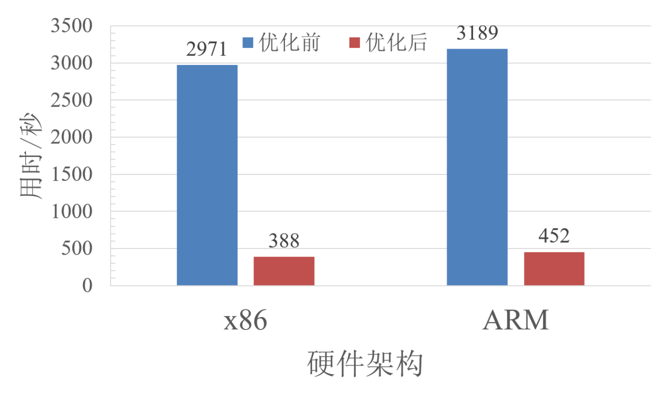


图 6 *fits_warp* 在 x86 和 ARM 上的优化效果

Figure 6 Parallel performance of *fits_warp* on x86 and ARM

表 3 软硬件配置

Table 3 Software and hardware configuration

指令集	CPU	内存	操作系统	内核版本	软件环境
x86	Intel Xeon Gold 6132 (28cores 2.60GHz) × 2	6-Channel DDR4-2666	CentOS 7	3.10.0	Python:2.7 gcc:8.3 wsclean:2.7
ARM	Kunpeng 920 CPU (48cores 2.50GHz) × 2	8-Channel DDR4-3200	CentOS 7	4.14.0	Python:2.7 gcc:8.3 wsclean:2.7

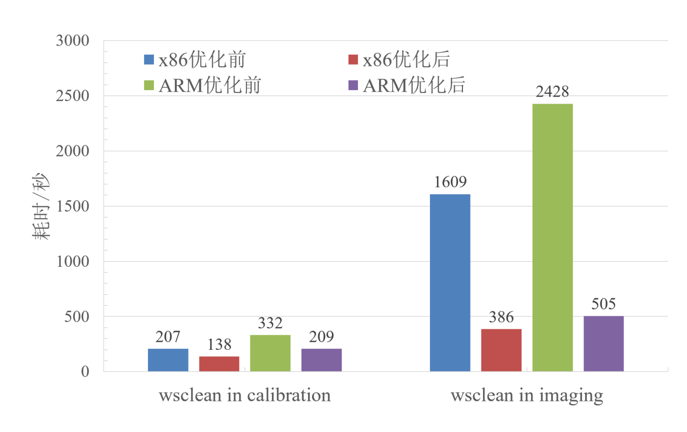


图 7 *wsclean* 在 x86 和 ARM 平台上的优化效果

Figure 7 Parallel performance of *wsclean* on x86 and ARM

3.1 *fits_warp* 优化结果

如图 6所示, *fits_warp* 的运行时间在优化后大幅下降, 在 x86 和 ARM 平台上分别获得 7.66 和 7.10 的加速比. 受限于每个批次 *fits_warp* 处理的文件数, 我们启动 12 个 Python 进程并行校准, 得出的并行效率分别为 $\frac{7.66}{12} = 63.8\%$ 和 $\frac{7.10}{12} = 58.8\%$, 接近于理论上的并行效率的表 2.

x86 和 ARM 平台上单个节点的 CPU 核心总数都较大, 分别是 28 和 96. 管线出于精度考虑设置的每批次文件数 (12), 制约了并行资源利用率的进一步提升. 未来的优化方向是在作业协调系统, 如 DALiuGE [56], 的协助下, 在一个计算节点上启动多条成像管线, 并行运行多个数据集管线.

3.2 *wsclean* 优化结果

wsclean 会根据调用它的阶段 (❸校准或❹成像执行不同的子函数. 我们在这两个阶段尝试不同的 *wsclean* 参数组合, 寻找最佳的参数. 在优化之前, −*j* 参数被设置为 CPU 核心的数量 (24). 优化后, 在❸校准阶段中把 −*j* 参数设置为 7, 并启动 4 个进程. 在❹成像阶段中, −*j* 被设置为 24, 并启动了 4 个进程. 参数变化和C++ 多线程优化的结合使得程序在 x86 和 ARM 平台上运行时间的下降, 其效果显示在图 7中. 原有的 *wsclean* 已经是并行运行的, 我们使用这个版本的运行时间作为计算加速比的基线. 在❸校准阶段, *wsclean* 在 x86 和 ARM 平台上分别实现了 1.5 和 1.6 的加速比. 在❹成像阶段, *wsclean* 在 x86 和 ARM 平台上分别获得了 4.2 和 5.0 的加速比, 展现了不错的性能提升. 分析表明, ❸校准阶段调用的子函数 *runFirstInversion* 的运行时间占主导地位, 该函数串行运行的比例较高 (70%-80%), 根据阿姆达尔定律, 预期能达到的加速比较低. 相对地, ❹成像阶段中的热点子函数 *predict* 和 *imageMain* 有很大一部分是可并行的, 根据阿姆达尔定律, 这两个子函数获得了较高的加速比. 优化后, *wsclean* 在这个阶段实现了 53.4% 和 43.1% 的并行效率, 低于表 2中的理论并行效率 65.5%. 进一步分析发现, *imageMain* 进行并行写优化之后, 内存带宽成为瓶颈. 这类访存密集应用的优化仍然是难点, 未来可考虑重新设计算法, 减少每次算术操作的访存次数.

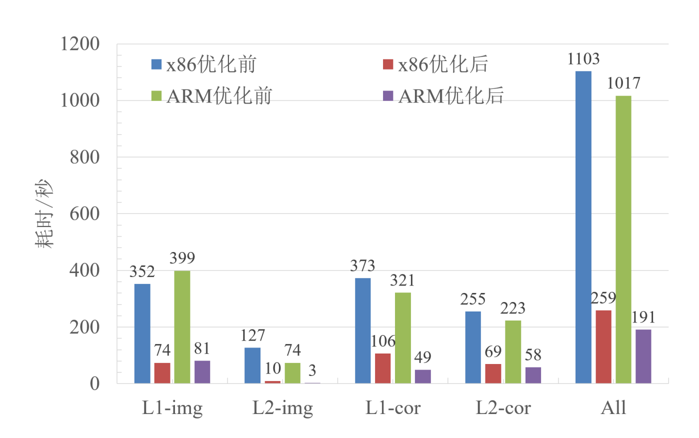


图 8 *image.sh* 和 *snapshot_correct.sh* 中四个循环的多进程结果.All 表示四个优化循环运行时间的总和.

Figure 8 Multiprocess results for the four loops in *image.sh* and *snapshot_correct.sh*.' All' represents the sum of the run times of the four optimization loops.

3.3 *shell_loops* 优化结果

如图 8所示, 在对 *image.sh* 和 *snapshot_correct.sh* 进行 Shell 多进程优化后, 运行时间大大减少. 这种优化方法在 x86 和 ARM 平台上都适用.

在加速比方面, L2-img 获得了最高加速比, 在 x86 和 ARM 平台上分别得到了 12.3 和 24.7 的加速比. 其他循环的加速比如下: L1-img分别获得了 4.8 和 5.0 的加速比; L1-cor分别获得了 3.5 和 6.6 的加速比; L2-cor分别获得了 3.7 和 3.8 的加速比; 整体上, 我们在 x86 和 ARM 平台上分别获得了 $\frac{1103}{259} = 4.26$ 和 $\frac{1017}{191} = 5.80$ 的加速比.

在并行效率方面, L1-img的并行效率接近 $\frac{5}{5} = 100\%$; L2-img在 x86 和 ARM 平台上的并行效率分别为 $\frac{12.3}{24} = 52.1\%$ 和 $\frac{24.7}{96} = 25.8\%$; L1-cor在 x86 和 ARM 平台上的并行效率分别为 $\frac{3.5}{8} = 43.8\%$ 和 $\frac{6.55}{8} = 81.9\%$. L2-cor在 x86 和 ARM 平台上的并行效率分别为 $\frac{3.69}{4} = 92\%$ 和 $\frac{3.84}{4} = 96\%$. 我们在运行时间、加速比和并行效率方面对 x86 和 ARM 平台进行了比较. 由于有更多的 CPU 内核, ARM 在所有循环的运行时间和加速比方面均优于 x86 平台. 只有L2-image 使用了所有 24 和 96 个 CPU 内核, 其他三个循环没有表现出足够的并行性. 相较于 ARM 在 CPU 核心数上的优势 ($\frac{96}{24} = 4.0$), 循环代码中相对较低的并行性制约了 ARM, 使其在运行时间上相比 x86 仅获得较小的优势 ($\frac{259}{191} = 1.4$).

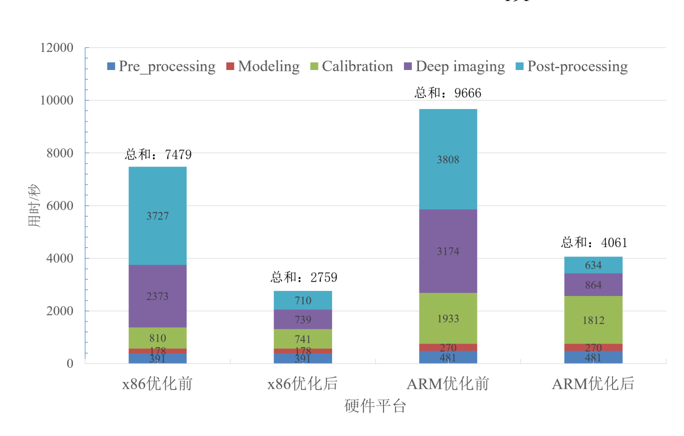


图 9　成像管线整体优化结果

Figure 9　Overall optimization results of the imaging pipeline

整个 MWA 成像管线的优化结果总结在图 9中. 管线的运行时间在 x86 上从 7479 秒降至 2759 秒, 在 ARM 上从 9666 秒降至 4016 秒, 分别获得了 2.7 和 2.4 的加速比. 在五个阶段当中, ❸校准阶段的运行时间在优化后降低了 5%. ❹成像和❺数据后处理阶段的运行时间大大下降了, 因为 *wsclean* 和 *shell_loops* 主要在这两个阶段执行.

3.4 验证成像结果

由于成像管线侧重于连续谱科学研究, 在这一节中我们主要比较和分析总强度 (Stokes I) 图像结果. 我们使用了射电源查找工具 *aegean*, 从原始和优化后的成像管线获得的图像上进行源搜寻并生成这些图像的射电源星表结果. 这些图像中搜寻到的射电源如表 4所示. 在表 4中, 我们使用 *Topcat* [40] 工具将所有星表结果与 x86 上运行的原始成像管线获得的图像的星表进行交叉匹配, 允许的最大误差设为 100 arcsec.

从表 4可以看出, 原始的管线在两种处理器上的结果大致相同, 只有近 0.7% 的差异. 优化后的管线获得的结果与原始管线获得的结果的匹配度分别为 97.6% 和 97.3%. 造成优化前后差异的原因是, 优化后的管线使用了并行反卷积 (–parallel-deconvolution 512) 和并行网格化 (–parallel-gridding 8). 在并行反卷积中, 使用 Dijkstra 算法将原始图像分割成许多最大尺寸为 512 的子图像, 该算法计算出子图像的最小分割路径, 并以所需的分割路径为界, 最小的分割路径在每次主循环的反卷积迭代之前都要重新确定 [41]. 在并行网格化中, 原始网格被分割成 8 个子网格并同时执行网格化. 这两种并行方法的使用, 将使得合并后的图像在子图像或子网格的临界位置出现轻微的混叠效应, 但这种影响被认为是可以忽略的 [42].

优化后的成像管线获得的峰值流量密度和位置的准确性对比分析如图 10所示. 图中, 峰值流量的比值 S_{x86_opt}/S_{x86_orig} 和 S_{ARM_opt}/S_{x86_orig} 的分布都集中在 1.0 左右, 说明优化后的流量密度与原始流量密度基本相等.x86 和 ARM 数据上的赤经位置差值 Δ_{RA} 的分布集中在-0.1 左右, x86 和 ARM 数

据上的赤纬位置差值 Δ_{DEC} 的分布集中在 0.0 左右, 表明优化后获得的位置与原来的基本相同. 这说明优化后的成像管线得到的结果是正确的, 优化方法是有效的.

表 4　成像结果对比
Table 4　Comparison of imaging results

平台	软件版本	检测数	匹配数
x86	原始版本	1,075	1,075
ARM	原始版本	1,075	1,068
x86	优化版本	1,077	1,049
ARM	优化版本	1,073	1,046

4　相关工作

随着 SKA 数据处理技术的发展, 天文学和高性能计算领域 [43–45] 的联系日趋紧密. 天文学应用的特点是数据规模大, 并行度高 [46–49]. SKA 完成一次巡天会产生海量的原始和图像数据, 分别在 PB 和 TB 量级. 天文学应用目前面临着计算资源消耗大、对计算资源依赖强的挑战. 因此, 天文数据处理流程的性能优化研究得到了广泛的关注. 近年来, 研究人员在 CPU、GPU、FPGA 等不同计算装置上对 SKA 先导应用设计了多种性能优化策略.

ARL 算法参考库是 SKA 天文学数据处理领域的一个重要算法库, 其中的关键算法 CLEAN 用于巡天观测中针对探测到的快照的成像工作. [50] 将原始算法移植到 GPU 上, 进行了并行加速, 并利用 M31 模拟数据对算法进行了分析与测试. 在不影响图像质量的前提下, 该优化使算法的运行耗时更短, 计算效率更高.

[51] 使用 SKA 观测数据研究了天文学数据处理中的两个关键算法: gridding 算法和 degridding 算法. 作者采用网格数据分块的方法, 通过循环展开、设置管线, 以及重新定制核心运算模块和存储器控制模块, 对算法实现了低功耗、高并行度的 FPGA 优化设计.

同时, 随着计算规模的扩大, 计算节点之间的数据通信开销也必须纳入考虑. [52] 基于已有的有向无环图任务集表示, 重点关注节点之间的海量通信, 即边缘通信的代价. 作者改进了现有的轮询方法, 将通信事件按成本划分为高低两种, 并针对二者设计了路径修正方案和边缘通信路径算法, 从而减少了观测数据处理系统的整体执行时间.

此外, 低频射电脉冲星搜索也是 SKA 的热点先导应用之一. 基于 FAST 观测数据, [53] 针对不同种类的脉冲星的运动特点进行了天空建模、天文数据处理以及红移效应的修正, 优化了脉冲星双星系统的搜索算法的性能以及并行时负载不平衡的问题. [54] 针对 PRESTO 软件套件, 在 CPU 上对消色散和离散快速傅里叶变换这两个子流程做了进程级并行, 取得了显著的加速效果.

5　总结

本文在中国 SKA 区域中心原型机上, 对 MWA 低频成像应用进行了并行性能优化. 通过热点剖析, 发现了应用整体流程的性能热点在于 *fits_warp.py*、*wsclean*、*calibrate* 和 *shell_loops* 函数. 其中, *calibrate* 在原有的并行方法下已经取得了较高的并行效率, 因此没有做进一步优化. 对于 *shell_loops* 和 *fits_warp.py*, 本文主要使用了 Python multiprocessing 进行了多进程的并行, 取得了很高的并行加速比. 对于 wsclean, 本文优化其C++多线程并行算法, 验证了算法精度能满足 SKA 使用要求. 本文的优化方法被证明在 x86 和 ARM 平台上都有很好的并行效果, 相比原先使用的版本, 分别实现了 2.7 和 2.4 倍整体加速. 同时, 本文的并行优化方案具有良好的扩展性, 对未来其他天文学应用的优化有很好的借鉴意义. 天文学应用往往有较好的并行性和庞大的数据量, 在未来的合理地将计算量分配到计算节点, 挖掘不同并行方法, 可以更好地提升数据的处理效率.

致谢　本研究得到了国家重点研发计划 (2018YFA040600, 2018YFA040603) 资助. 本研究使用了由国家重点研发计划大科学装置前沿研究专项 (2018YFA0404603) 和中国科学院 (114231KYSB20170003) 资助研制的中国 SKA 区域中心原型机的资源.

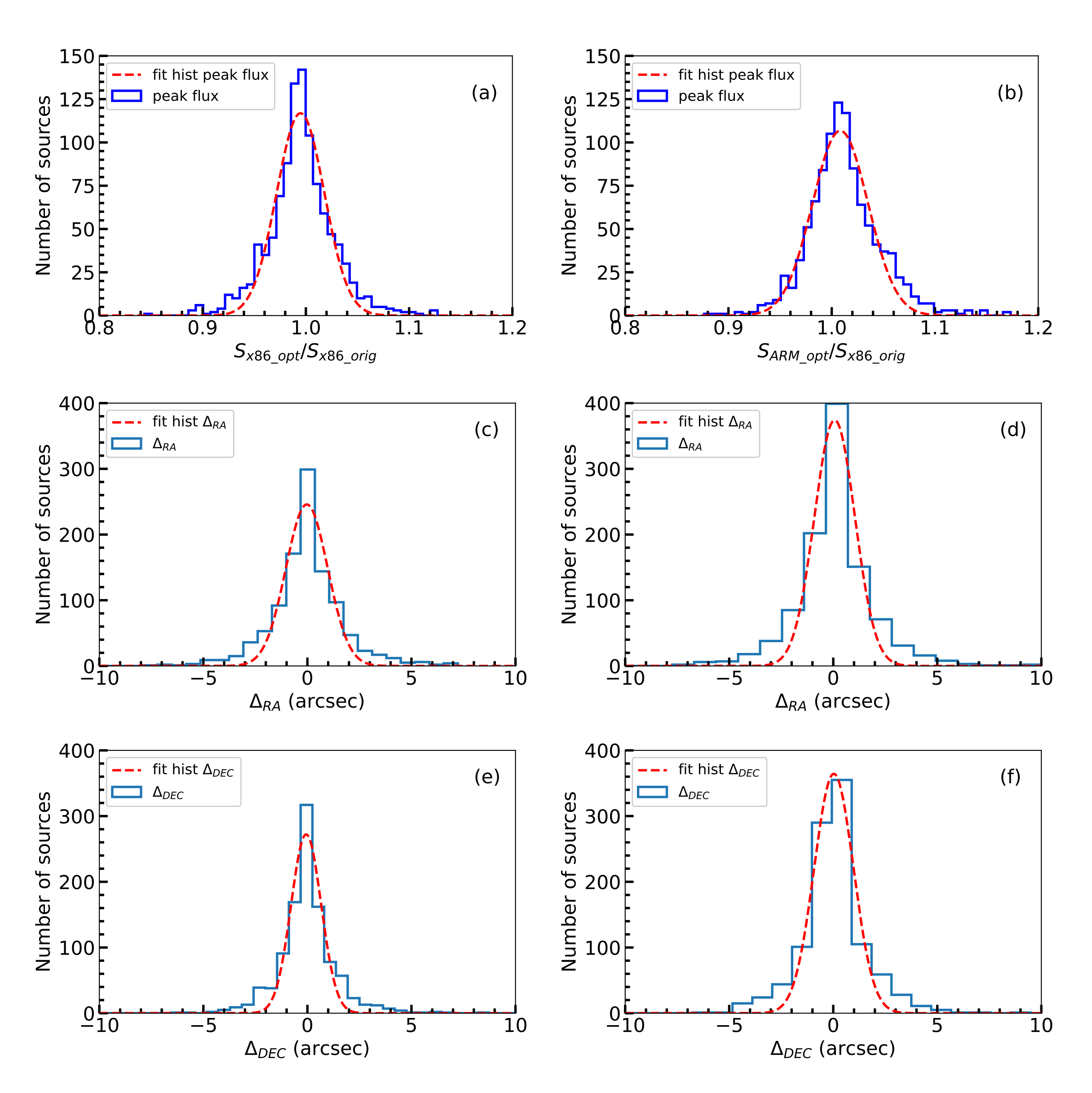


图 10 交叉匹配源的统计分析结果. 图 (a) 和 (b) 分别是优化后管线在 x86 和 ARM 上运行获得的峰值流量密度与原始管线在 x86 上获得的比值分布图.(a) 中的拟合结果: 平均值为 1.0, 振幅为 116.9, 标准差为 0.02; (b) 中的拟合结果: 平均值为 1.0, 振幅为 106.7, 标准差为 0.03. 图 (c) 和 (d) 分别是优化后管线在 x86 和 ARM 上获得的射电源的赤经 RA 值与原始管线在 x86 上获得的 RA 值的差值 Δ_{RA} 分布图. (c) 中的拟合结果: 平均值为-0.03, 振幅为 245.8, 标准差为 1.0; (d) 中的拟合结果: 平均值为 0.07, 振幅为 374.1, 标准差为 1.0. 图 (e) 和 (f) 分别是来优化后管线在 x86 和 ARM 上获得的射电源的赤纬 DEC 值与原始管线在 x86 上获得的射电源的 DEC 的差值 Δ_{DEC} 分布图.(e) 中的拟合结果: x86 上的平均值为-0.06, 振幅为 272.2, 标准差为 0.7; (f) 中的拟合结果: 平均值为 0.03, 振幅为 364.3, 标准差为 0.9.

Figure 10 The statistical analysis results of cross match sources. The top two maps are the peak flux density from optimization pipeline on x86 and ARM ratio to those from original pipeline on x86, respectively. The peak flux density gaussian fitted parameters are: mean is 1.0, amplitude is 116.9, standard deviation is 0.02 on x86, and mean is 1.0, amplitude is 106.7, standard deviation is 0.03 on ARM. The middle two maps are right ascension (RA) from is from optimization pipeline on x86 and ARM subtracted those from the original pipeline on x86. The Δ_{RA} gaussian fitted parameters are: mean is -0.03, amplitude is 245.8, standard deviation is 1.0 on x86, and mean is 0.07, amplitude is 374.1, standard deviation is 1.0 on ARM. The bottom two maps are declination (DEC) from is from optimization pipeline on x86 and ARM subtracted those from the original pipeline on In tel6132. The Δ_{DEC} gaussian fitted parameters are: mean is -0.06, amplitude is 272.2, standard deviation is 0.7 on x86, and mean is 0.03, amplitude is 364.3, standard deviation is 0.9 on ARM.

参考文献

1 I. Prandoni, and N. Seymour, Revealing the Physics and Evolution of Galaxies and Galaxy Clusters with SKA Continuum Surveys. Advancing Astrophysics with the Square Kilometre Array (AASKA14), 2015,: 67
2 Van Eck C L, Haverkorn M, Alves M I R, et al. Astron. Astrophys, 2019, 623: A71.
3 Riseley C J, Galvin T J, Sobey C, et al. Publ. Astron. Soc. Aust, 2020, 37: E029.
4 Hale C L, McConnell D, Thomson A J M, et al. Publ. Astron. Soc. Aust, 2021, 38: e058.
5 Botteon A, Shimwell T W, Cassano R, et al. Astron. Astrophys, 2022, 660: A78.
6 Norris R P, Marvil J, Collier J D, et al. Publ. Astron. Soc. Aust, 2021, 38: E046.
7 Y. M. Wang, A. Tuntsov, T. Murphy, E. Lenc, M. Walker, K. Bannister, D. L. Kaplan, and E. K. Mahony, Mon. Not. R. Astron. Soc, 502, 3294 (2021), arXiv:2101.06048.
8 J. F. Helmboldt, and N. Hurley-Walker, Radio Science, 55, 1 (2020).
9 R. H. Becker, R. L. White, and D. J. Helfand, The Astrophysical Journal, 450, 559 (1995)
10 J. J. Condon, W. D. Cotton, E. W. Greisen, Q. F. Yin, R. A. Perley, G. B. Taylor, and J. J. Broderick, The Astronomical Journal, 115, 1693 (1998)
11 Lacy, M., Baum, S. A., Chandler, C. J., et al. The Karl G. Jansky Very Large Array Sky Survey (VLASS). Science Case and Survey Design. Publications of the Astronomical Society of the Pacific, 2020, 132: 035001
12 H. T. Intema, P. Jagannathan, K. P. Mooley, and D. A. Frail, Astron. Astrophys, 598, A78 (2017)
13 R. B. Wayth, E. Lenc, M. E. Bell, J. R. Callingham, K. S. Dwarakanath, T. M. O. Franzen, B.-Q. For, B. Gaensler, P. Hancock, L. Hindson, N. Hurley-Walker, C. A. Jackson, M. Johnston-Hollitt, A. D. Kapinska, B. McKinley, J. Morgan, A. R. Offringa, P. Procopio, L. Staveley-Smith, C. Wu, Q. Zheng, C. M. Trott, G. Bernardi, J. D. Bowman, F. Briggs, R. J. Cappallo, B. E. Corey, A. A. Deshpande, D. Emrich, R. Goeke, L. J. Greenhill, B. J. Hazelton, D. L. Kaplan, J. C. Kasper, E. Kratzenberg, C. J. Lonsdale, M. J. Lynch, S. R. McWhirter, D. A. Mitchell, M. F. Morales, E. Morgan, D. Oberoi, S. M. Ord, T. Prabu, A. E. E. Rogers, A. Roshi, N. Udaya Shankar, K. S. Srivani, R. Subrahmanyan, S. J. Tingay, M. Waterson, R. L. Webster, A. R. Whitney, A. Williams, and C. L. Williams, Publ. Astron. Soc. Aust, 32, e025 (2015), arXiv:1505.06041.
14 T. W. Shimwell, H. J. A. Röttgering, P. N. Best, W. L. Williams, T.J. Dijkema, F. de Gasperin, M. J. Hardcastle, G. H. Heald, D. N. Hoang, A. Horneffer, H. Intema, E. K. Mahony, S. Mandal, A. P. Mechev, L. Morabito, J. B. R. Oonk, D. Rafferty, E. Retana-Montenegro, J. Sabater, C. Tasse, R. J. van Weeren, M. Brüggen, G. Brunetti, K. T. Chyży, J. E. Conway, M. Haverkorn, N. Jackson, M. J. Jarvis, J. P. McKean, G. K. Miley, R. Morganti, G. J. White, M. W. Wise, I. M. van Bemmel, R. Beck, M. Brienza, A. Bonafede, G. Calistro Rivera, R. Cassano, A. O. Clarke, D. Cseh, A. Deller, A. Drabent, W. van Driel, D. Engels, H. Falcke, C. Ferrari, S. Fröhlich, M. A. Garrett, J. J. Harwood, V. Heesen, M. Hoeft, C. Horellou, F. P. Israel, A. D. Kapińska, M. Kunert-Bajraszewska, D. J. McKay, N. R. Mohan, E. Orrú, R. F. Pizzo, I. Prandoni, D. J. Schwarz, A. Shulevski, M. Sipior, D. J. B. Smith, S. S. Sridhar, M. Steinmetz, A. Stroe, E. Varenius, P. P. van der Werf, J. A. Zensus, and J. T. L. Zwart, Astron. Astrophys, 598, A104 (2017), arXiv:1611.02700.
15 M. J. Jarvis, A. R. Taylor, I. Agudo, J. R. Allison, R. P. Deane, B. Frank, N. Gupta, I. Heywood, N. Maddox, K. McAlpine, M. G. Santos, A. M. M. Scaife, M. Vaccari, J. T. L. Zwart, E. Adams, D. J. Bacon, A. J. Baker, B. A. Bassett, P. N. Best, R. Beswick, S. Blyth, M. L. Brown, M. Bruggen, M. Cluver, S. Colafranceso, G. Cotter, C. Cress, R. Dave, C. Ferrari, M. J. Hardcastle, C. Hale, I. Harrison, P. W. Hatfield, H. R. Klockner, S. Kolwa, E. Malefahlo, T. Marubini, T. Mauch, K. Moodley, R. Morganti, R. Norris, J. A. Peters, I. Prandoni, M. Prescott, S. Oliver, N. Oozeer, H. J. A. Rottgering, N. Seymour, C. Simpson, O. Smirnov, D. J. B. Smith, K. Spekkens, J. Stil, C. Tasse, K. van der Heyden, I. H. Whittam, W. L. WIlliams, (2017), arXiv:1709.01901
16 R. P. Norris, A. M. Hopkins, J. Afonso, S. Brown, J. J. Condon, L. Dunne, I. Feain, R. Hollow, M. Jarvis, M. Johnston-Hollitt, E. Lenc, E. Middelberg, P. Padovani, I. Prandoni, L. Rudnick, N. Seymour, G. Umana, H. Andernach, D. M. Alexander, P. N. Appleton, D.Bacon, J. Banfield, W. Becker, M. J. I. Brown, P. Ciliegi, C. Jackson, S. Eales, A. C. Edge, B.M. Gaensler, G. Giovannini, C. A. Hales, P. Hancock, M. Y. Huynh, E. Ibar, R. J. Ivison, R. Kennicutt, A. E. Kimball, A. M. Koekemoer, B. S. Koribalski, Á. R. López-Sánchez, M. Y. Mao, T. Murphy, H. Messias, K. A. Pimbblet, A. Raccanelli, K. E. Randall, T. H. Reiprich, I. G. Roseboom, H. Röttgering, D. J. Saikia, R. G. Sharp, O. B. Slee, I. Smail, M. A. Thompson, J. S. Urquhart, J. V. Wall, and G. B. Zhao, Publ. Astron. Soc. Aust, 28, 215 (2011), arXiv:1106.3219.
17 N. Hurley-Walker, J. R. Callingham, P. J. Hancock, T. M. O. Franzen, L. Hindson, A. D. Kapinska, J. Morgan, A. R. Offringa, R. B. Wayth, C. Wu, Q. Zheng, T. Murphy, M. E. Bell, K. S. Dwarakanath, B. Q. For, B. M. Gaensler, M. Johnston-Hollitt,

E. Lenc, P. Procopio, L. Staveley-Smith, R. Ekers, J. D. Bowman, F. Briggs, R. J. Cappallo, A. A. Deshpande, L. Greenhill, B. J. Hazelton, D. L. Kaplan, C. J. Lonsdale, S. R. McWhirter, D. A. Mitchell, M. F. Morales, E. Morgan, D. Oberoi, S. M. Ord, T. Prabu, N. U. Shankar, K. S. Srivani, R. Subrahmanyan, S. J. Tingay, R. L. Webster, A. Williams, and C. L. Williams, Mon. Not. R. Astron. Soc, 464, 1146 (2017), arXiv:1610.08318.

18 N. Hurley-Walker, M. D. Filipovic, B. M. Gaensler, D. A. Leahy, P. J. Hancock, T. M. O. Franzen, A. R. Offringa, J. R. Callingham, L. Hindson, C. Wu, M. E. Bell, B. Q. For, M. Johnston-Hollitt, A. D. Kapinska, J. Morgan, T. Murphy, B. McKinley, P. Procopio, L. Staveley-Smith, R. B. Wayth, and C. Q. Zheng, Publ. Astron. Soc. Aust, 36, e045 (2019), arXiv:1911.08126

19 S. V. White, T. M. O. Franzen, C. J. Riseley, O. I. Wong, A. D. Kapińska, N. Hurley-Walker, J. R. Callingham, K. Thorat, C. Wu, P. Hancock, R. W. Hunstead, N. Seymour, J. Swan, R. Wayth, J. Morgan, R. Chhetri, C. Jackson, S. Weston, M. Bell, B. Q. For, B. M. Gaensler, M. Johnston-Hollitt, A. Offringa, and L. Staveley-Smith, Publ. Astron. Soc. Aust, 37, e018 (2020), arXiv:2004.13125.

20 Riseley, C. J., Lenc, E., Van Eck, C. L., et al. The POlarised GLEAM Survey (POGS) I: First results from a low-frequency radio linear polarisation survey of the southern sky. Publications of the Astronomical Society of Australia, 2018, 35: 43

21 Riseley, C. J., Galvin, T. J., Sobey, C., et al. The POlarised GLEAM Survey (POGS) II: Results from an all-sky rotation measure synthesis survey at long wavelengths. Publications of the Astronomical Society of Australia, 2020, 37: e029

22 George, L. T., Dwarakanath, K. S., Johnston-Hollitt, M., et al. A study of halo and relic radio emission in merging clusters using the Murchison Widefield Array. Monthly Notices of the Royal Astronomical Society, 2017, 467: 936

23 For, B.-Q., Staveley-Smith, L., Hurley-Walker, N., et al. A multifrequency radio continuum study of the Magellanic Clouds - I. Overall structure and star formation rates. Monthly Notices of the Royal Astronomical Society, 2018, 480: 2743

24 Tremblay, C. D., Jones, P. A., Cunningham, M., et al. A Molecular Line Survey around Orion at Low Frequencies with the MWA. The Astrophysical Journal, 2018, 860: 145

25 H. Y. Gong, Z. L. Zhang, M. Y. Xue, X. Y. Hong, T. An, X. Z Hu, Z. J. Xu, S. G. Guo, B. Q. Lao, X. C Wu, and T. Yu, SCIENTIA SINICA Physica, Mechanica & Astronomica, 50, 109501 (2020).

26 T. An, X. P. Wu, and X. Y. Hong, SKA data take centre stage in China, Nature Astronomy, 3, 1030 (2019).

27 T.An, X.C.Wu, B.Q.Lao, et al. Status and progress of China SKA Regional Centre prototype. arxiv:2206.13022

28 B.Q.Lao, Y.K.Zhang, T.An, et al. Software Platform on China SKA Regional Center Prototype System(in Chinese).ChinaXiv:202206.00173.

29 J.W.Wei, C.F.Zhang, Z.L.Zhang, et al. Parallel optimization of the pulsar search pipeline on China SKA Regional Centre Prototype (in Chinese). ChinaXiv:T202206.00297.

30 T. An, Science opportunities and challenges associated with ska big data, SCIENCE CHINA Physics, Mechanics & Astronomy, 2019.

31 B. Q. Lao and T. An. Deployment of SKA low frequency imaging system in China SKA Regional Centre. SCIENTIA SINICA Physica, Mechanica & Astronomica. 50, 059501 (2020).

32 A. R. Offringa. Compression of interferometric radio-astronomical data. Astronomy & Astrophysics. 595, A99 (2016).

33 A. R. Offringa, R. B. Wayth, N. Hurley-Walker, et al. The low-frequency environment of the Murchison Widefield Array: radio-frequency interference analysis and mitigation. Publications of the Astronomical Society of Australia. 32, e008 (2015).

34 A. R. Offringa, C. M. Trott, N. Hurley-Walker, et al. Parametrizing Epoch of Reionization foregrounds: a deep survey of low-frequency point-source spectra with the Murchison Widefield Array. Monthly Notices of the Royal Astronomical Society. 458, 1057 (2016).

35 A. R. Offringa, B. McKinley, N. Hurley-Walker, F. H. Briggs, R. B. Wayth, D. L. Kaplan, M. E. Bell, L. Feng, A. R. Neben, J. D. Hughes, J. Rhee, T. Murphy, N. D. R. Bhat, G. Bernardi, J. D. Bowman, R. J. Cappallo, B. E. Corey, A. A. Deshpande, D. Emrich, A. Ewall-Wice, B. M. Gaensler, R. Goeke, L. J. Greenhill, B. J. Hazelton, L. Hindson, M. Johnston-Hollitt, D. C. Jacobs, J. C. Kasper, E. Kratzenberg, E. Lenc, C. J. Lonsdale, M. J. Lynch, S. R. McWhirter, D. A. Mitchell, M. F. Morales, E. Morgan, N. Kudryavtseva, D. Oberoi, S. M. Ord, B. Pindor, P. Procopio, T. Prabu, J. Riding, D. A. Roshi, N. Udaya Shankar, K. S. Srivani, R. Subrahmanyan, S. J. Tingay, M. Waterson, R. L. Webster, A. R. Whitney, A. Williams, and C. L. Williams, Mon. Not. R. Astron. Soc, 444, 606 (2014).

36 N. Hurley-Walker, and P. J. Hancock, Astron. Comput, 25, 94 (2018), arXiv: 1808.08017.

37 M. I. Large, L. E. Cram and A. M. Burgess. A machine-readable release of the Molonglo Reference Catalogue of Radio Sources,

The Observatory. 111, 72 (1991).

38 J. J. Condon, W. D. Cotton, E. W. Greisen, et al. The NRAO VLA sky survey. Astronomical Journal. 115, 1693 (1998).

39 Hancock, P. J., Trott, C. M., and Hurley-Walker, N. (2018). Source finding in the era of the SKA (Precursors): Aegean 2.0. Publications of the Astronomical Society of Australia, 35.

40 M. B. Taylor, (2005, December). TOPCAT & STIL: Starlink table/VOTable processing software. In Astronomical data analysis software and systems XIV (Vol. 347, p. 29).

41 B. Veenboer, S. Van der Tol, A. R. Offringa, J. W. Romein and T. J. Dijkema, Radio-Astronomical Imaging with WSClean and Image-Domain Gridding. In 2020 XXXIIIrd General Assembly and Scientific Symposium of the International Union of Radio Science (pp. 1-4). IEEE.

42 A. R. Offringa, F. Mertens, S. van der Tol, B. Veenboer, B. K. Gehlot, L. V. E. Koopmans, and M. Mevius, Astron. Astrophys, 631, A12 (2019), arXiv:1908.11232.

43 L G. CHEN, R. MAO, Z. K. LU, Chinese Science Bulletin, 5, 566 (2015).

44 F. H. SHEN, W. S. LUO, H. ZHAO, Application Research of Computers, 2, 52 (2004).

45 BRENNAN J, KURESHI I, HOLMES V. CDES: an approach to HPC workload modelling. Proceedings of International Symposium on IEEE/ACM 18th Distributed Simulation and Real Time Applications, 2014, 47-54.

46 Y. ZHANG, and Y. ZHAO, Data Science Journal, 14, 1 (2015).

47 T. An, Science China Physics, Mechanics, and Astronomy, 62, 989531 (2019), arXiv:1901.07756.

48 Guo S, Zheng X, Mao Y, et al. Scheme and Prospect of the SKA Big Data Transferring. Frontiers of Data & Computing, 2018, 9(03), 3-13.

49 WANG Y, LI P, JI Y, et al. High Performance Computing and Astronomical Data:A Survey. Computer Science, 2020,47(01),1-6.

50 Liu H. Optimization research on CLEAN algorithm in ARL. Zhengzhou University of Light Industry, 2019.

51 Wu Q. Design and optimization of high efficiency hardware accelerator for the key algorithm of SKA science data processing. Shanghai Jiao Tong University, 2019.

52 Xue Q. Research on science data processing scheduling for SKA science data processing. Shanghai Jiao Tong University, 2019.

53 C. Ji, Research on speed up and optimization of pulsar acceleration search in frequency domain. Guizhou Normal University, 2021.

54 Q. Y. Yu, Z. C. Pan, L. Qian, S. Wang, Y. L. Yue, M. L. Huang, Q. L. Hao, S. P. You, B. Peng, Y. Zhu, L. Zhang, and Z. J. Liu, Res. Astron. Astrophys, 20, 091 (2020).

55 O. Tange, The USENIX Magazine, 36, 42 (2011).

56 Wu, Chen, et al. "DALiuGE: A graph execution framework for harnessing the astronomical data deluge." Astronomy and computing 20 (2017): 1-15.

57 Gustafson, John L. "Reevaluating Amdahl's law." Communications of the ACM 31.5 (1988): 532-533.

Optimization of parallel processing of Square Kilometre Array low frequency imaging pipeline

Jianwen Wei[1], Chenfei Zhang[1], Baoqiang Lao[2,3]*, James Lin[1] & Tao An[2]

1. Shanghai Jiao Tong University, Shanghai, 200240;
2. Shanghai Astronomical Observatory, Chinese Academy of Sciences, Shanghai, 200030;
3. School of Physics and Astronomy, Yunnan University, kunming, 650500

Data processing of the Square Kilometre Array (SKA) is carried out in pipeline mode, and the execution efficiency of pipeline is an important factor to be considered in SKA data processing. Continuum imaging is one of the main observation modes of SKA and is a prerequisite for many other scientific works. In this paper, we take the imaging pipeline of SKA low-frequency precursor Murchison Widefield Array (MWA), as an example and optimize the parallel processing pipeline on the China SKA Regional Centre prototype (CSRC-P). Previous optimization schemes have focused on a few performance hotspots and lacked systematic optimization of the overall pipeline, resulting in a relatively poor overall speedup ratio. In this paper, we propose a global optimization scheme that combines C++ multi-threading, Python multi-processing, and Shell multi-tasking parallelism for pipelines using multiple programming languages and image datasets that can be processed independently, and verify the accuracy of the optimization results. Experiments show taht the optimized pipeline achieves an overall speedup of 2.7 and 2.4 times on the x86 and ARM nodes of CSRC-P, respectively, and the ARM compute nodes shows good adaptability to SKA applications. The optimization strategies and methods in this paper are also applicable to other SKA applications and will be useful for the scientific operation and future operation of the SKA precursor telescope.

Square Kilometre Array, Murchison Widefield Array, high performance computing, parallel optimization

PACS: 47.27.-i, 47.27.Eq, 47.27.Nz, 47.40.Ki, 47.85.Gj

doi: ??